UNITED STATES PATENT AND TRADEMARK OFFICE

A

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/694,282 | 10/27/2003 | Ian S. Eslick | 51346/PAN/B600 | 2759 |

23363 7590 09/16/2005

CHRISTIE, PARKER & HALE, LLP
PO BOX 7068
PASADENA, CA 91109-7068

| EXAMINER | |
|---|---|
| COLEMAN, ERIC | |
| **ART UNIT** | **PAPER NUMBER** |
| 2183 | |

DATE MAILED: 09/16/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 10/694,282 | ESLICK ET AL. | |
| | Examiner | Art Unit | |
| | Eric Coleman | 2183 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) 1-30 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-30 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

## DETAILED ACTION

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

Claims 1-30 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-32 of U.S. Patent No. 6,675,689 (hereafter referred to as patent '689. Although the conflicting claims are not identical, they are not patentably distinct from each other because of the following:

Claims in patent '689 correspond the clams in the instant application as follows:

| Instant application | Patent 6,675,689 |
|---|---|
| Claim 1. | Claim 1. |
| As method of executing software, comprising: | As method of executing software, comprising: |

identifying a first configuration information required to execute a first kernel code segment;

Selecting a first accelerator set configured to execute said first kernel code segment ; and initiating a direct memory access transfer of said first configuration information to said first accelerator set.

Claim 2 the method of claim 1 further comprising building an entry in a kernel code execution table utilizing said first kernel code segment and said first configuration information.

Claim 3. the method of claim 1 wherein said first configuration information is stored in a register and wherein the register defines a context.

Claim 4. the method of claim 1, further comprising identifying a first set of arguments.

retrieving a first kernel segment ;

identifying a first configuration information required to execute said first kernel code segment;

selecting a first accelerator set configured to execute said first kernel code segment; and initiating a direct memory access transfer to said first accelerator set;

building a entry in a kernel code execution table utilizing said first kernel code segment and said first configuration information.

Claim 2 wherein said first configuration information is stored in a register.

Claim 3 the method of claim 1 wherein the register defines a context.

Claim 4. the method of claim 1, further comprising identifying a first set of arguments.

Claim 5. The method of claim 4 wherein said initiating said direct memory access transfer further comprises transferring said first set of arguments.

Claim 6. The method of claim 1, further comprising identifying a first set of microcode.

Claim 7. The method of claim 6 wherein said initiating said direct memory access transfer further comprises transferring said first set of microcode.

Claim 8. the method of claim 1, further comprising identifying a second configuration information required to execute a second kernel code segment and selecting a second accelerator set configured to execute said second kernel code segment.

[

]

Claim 5. the method of claim 4, wherein said initiating said direct memory access transfer further includes transferring said first set of arguments.

Claim 6. The method of claim 1, further comprising identifying a first set of microcode.

Claim 7. The method of claim 6 wherein said initiating said direct memory access transfer further includes transferring said first set of microcode.

Claim 8 the method of claim 1 wherein initiating said direct memory access transfer includes transferring said first configuration information.

Claim 9, method of claim 1 further comprising retrieving a second kernel code segment.

Claim 10. the method of claim 1, further comprising selecting a second accelerator set.

Claim 9 the method of claim of claim 8 wherein the first accelerator set and said second accelerator set are overlapping.

[]

Claim 10 the method of claim 10 wherein said first accelerator set and said second accelerator set are non-overlapping.

Claim 11 the method of claim 10, wherein said first kernel code segment executes on said first accelerator set and said second kernel code executes on said second accelerator set concurrently with the first.

Claim 12. the method of claim 11, wherein the third kernel code segment executes on said first accelerator set subsequently to said first kernel code segment and concurrently with said second kernel code segment.

]

Claim 11. The method of claim 10, wherein said first accelerator set and said second accelerator set are overlapping.

Claim 12, the method of claim 10, wherein said first accelerator set and the second accelerator set are non-overlapping.

Claim 13, the method of claim 12, wherein said first kernel code segment executes on said accelerator set and said second kernel code segment executes on said second accelerator set concurrently.

Claim 14. the method of claim 13 wherein a third kernel code segment executes on said first accelerator set subsequent to said first kernel code Segment and concurrently with said second kernel code segment.

]

Claim 13. the method of claim 1 further comprising identifying a second configuration information required to execute a second kernel code segment, wherein initiating said direct memory access includes transferring the second configuration information to said first accelerator set while said first kernel code executes on said first accelerator set.

Claim 14. method of claim 1 wherein initiating said direct memory access includes transferring a first set of microcode to said accelerator set while said first kernel code set executes on said first accelerator set.

]

]

Claim 15. the method of claim 1 further comprising retrieving a second kernel code segment, wherein direct memory access includes transferring a first set of

Claim 15. the method of claim 9, wherein initiating said direct memory access includes transferring a second configuration information to said first accelerator set while said first kernel code set executes on said first accelerator set.

[

]

Claim 16 method of claim 9 wherein initiating said direct memory access includes transferring a first set of microcode to said first accelerator set while said first kernel code executes on said first accelerator set.

Claim 17. the method of claim 9, wherein initiating said direct memory access includes transferring first set of arguments to said first accelerator set

arguments to said first accelerator set while said first kernel code set executes on said first accelerator set.

Claim 16. The method of claim 1, further comprising determining completion requirements of said first kernel code segment to determine the order of execution of said first kernel code segment.

Claim 17. The method of claim 16, wherein said determining completion requirements of said first kernel code segment includes determining a variant of said first kernel code segment.

Claim 18. The method of claim 16, wherein said determining completion requirements of said first kernel code segment includes determining whether to execute said first kernel code segment in said accelerator set or in a second accelerator set.

while said first kernel code set executes on said first accelerator set.

]

Claim 18. the method of claim 1, further comprising determining completion requirements of said first kernel code segment to determine the order of execution of said first kernel code segment.

Claim 19, the method of claim 18, wherein said determining completion requirements of said first kernel code segment includes determining a variant of said first kernel code segment.

Claim 20, the method of claim 18, wherein said determining completion requirements of said first kernel code segment includes determining whether to execute said first kernel code segment in said accelerator set or in a second accelerator set.

[

Claim 19. An apparatus, comprising: a memory storing a set of configuration information; an accelerator coupled to the memory; and a kernel processor coupled to the memory, said kernel processor controlling the processing of at least one thread of program code on the accelerator by initiating a direct memory access controller of the configuration information to the accelerator.

Claim 20, the apparatus of claim 19, further comprising at least one main processor and memory, least one main processor being configured to process overhead code.

Claim 21. The apparatus of claim 19, wherein said the accelerator is a multiple context processing element.

Claim 22. The apparatus of claim 21, wherein said multiple context processing are elements grouped into overlapping

Claim 21. an apparatus comprising a memory storing at least one set of configuration information, the at least one set of configuration information describing at least one set of contexts;

At least one accelerator; and

A kernel processor coupled to the memory, said kernel processor controlling the processing of at least one thread of program code on said at least one set of configuration information.

Claim 22. The apparatus of claim 21, further comprising at least one main processor, configured to process overhead code.

[]

Claim 23. The apparatus of claim 21, wherein said at least one accelerator is a multiple context processing element.

Claim 24. The apparatus of claim 23 wherein said multiple context processing elements are grouped into overlapping

Bins.

Claim 23 the apparatus of claim 21, wherein said multiple context processing elements are grouped into non-overlapping bins.

Claim 24. the apparatus of claim 23 wherein said kernel processor is configured to load a first kernel code segment into a first one of said non-overlapping bins and to load a second kernel segment into a second one of said non-overlapping bins.

Claim 25. the apparatus of claim19, wherein the accelerator is a digital signal processor.

Claim 26. The apparatus of claim 25, wherein the digital signal processor has a single instruction cache.

Claim 27. The apparatus of claim 25, wherein the digital signal processor has dual instruction caches.

Bins.

Claim 25, The apparatus of claim 23, wherein said multiple context processing elements are grouped into non-overlapping bins.

Claim 26. The apparatus of claim 25, wherein said kernel processor is configured to load a first kernel code segment into a first one of said non-overlapping bins and to load a second kernel code segment into a second one of said non-overlapping bins.

Claim 27. the apparatus of claim 21, wherein said at least one accelerator is a digital signal processor.

Claim 28. The apparatus of claim 27, wherein said digital signal processor has a single instruction cache.

Claim 29. the apparatus of claim 27 wherein said digital signal processor has dual instruction caches.

Claim 28. the apparatus of claim 25, wherein the digital signal processor has an instruction cache configured with dual port memory, wherein a first port is coupled to a first bus and a second port is coupled to a second bus.

Claim 29. An apparatus configured to execute software comprising; means for identifying a first configuration information required to execute a first kernel code segment; means for selecting a first accelerator set configured to execute said first kernel code segment; and means for initiating a direct memory access transfer of said first configuration information to said first accelerator set.

[

]

Claim 30. the apparatus of claim 27, wherein said digital signal processor has an instruction cache configured with dual-port memory, wherein a first port is coupled to a first bus and a second port is coupled to a second bus.

Claim 31, An apparatus configured to execute software, comprising: means for retrieving a first kernel code segment; means for identifying a first configuration information required to execute said first kernel code segment; means for building an entry in a kernel code execution table utilizing said first kernel code segment and said first configuration information; means for selecting a first accelerator set configured to execute said first kernel segment; and means for initiating a direct memory access transfer to said first accelerator set.

Claim 30. A machine-readable medium having stored thereon instructions for processing elements, which when executed by said processing elements perform the following:

[

]

identifying a first configuration information required to execute a first kernel code segment; selecting a first accelerator set configured to execute said first kernel code segment; and initiating a direct memory access transfer of said first configuration information to said first accelerator set.

Claim 32. a machine readable medium having stored thereon instructions for processing elements which when executed by said processing elements perform the following: retrieving a first kernel code segment; identifying a first configuration information required to execute said first kernel code segment; building an entry in a kernel code execution table utilizing said first kernel code segment and said first configuration information; selecting a first accelerator set configured to execute said first kernel code segment; and initiating a direct memory access transfer to said first accelerator set.

From the side by side view of the corresponding claims of the patent 6,675,289

And the instant claims above, it can be readily seen that the inventions are substantially the same. As to the differences in claim 1, the patent contains the limitation "retrieving a first kernel segment" and "transfer of said first configuration information to said first accelerator set". As to the first limitation both the application claims and the patent claims use the first kernel segment to build a table. Therefore one of ordinary skill would have been motivated to retrieve the kernel segment in order build the table. As to the transfer of the first configuration information the patent uses the configuration information with the first kernel code segment and therefore one of ordinary skill would have been motivated to transfer the configuration information in order to have the configuration for later the claimed use.

As to claim 19 of the instant application, the limitation that the configuration information relates to contexts is contained in claim 21 of the patent but not in claim 19 of the application. However since claim 21 of the current application uses plural contexts then one of ordinary skill would have been motivated to provide that the configuration information of claim 19 contain contextual information for use in controlling the contexts the claim 19 as it relates to claim 21 of the instant application.

As to claims 29, and 30, these claims are similar to claim 32 of the patent. However the patent claim 32 contains additional limitations including retrieving a first kernel code segment building an entry in a kernel code execution table utilizing said first kernel code segment and said first configuration information. The instant application claim 30 selects accelerators for executing the kernel code. Therefore One of ordinary

skill would a have been motivated to retrieve the kernel code for use in the execution using the selected accelerator. Also one of ordinary skill would have been motivated to build an kernel code execution table to facilitate the of the kernel code segment for quick access to the needed data for execution.

As to the dependent claims as shown in the side to side listing of the claims above it is readily seen that the inventions are the same.

### *Claim Rejections - 35 USC § 112*

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claim 10 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The scope of meaning of claim 10 is unclear because claim 10 depends upon itself.

### *Specification*

The applicant is reminded that the cross-reference to related application(s) section of the application on page 1 of the specification should be updated as appropriate.

***Conclusion***

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Lichtman (patent No. 5,787,246) disclosed a system for configuring devices for a computer system (e.g., see abstract).

Dupree (patent No. 5,949,994) disclosed a dedicted context-cycling computer with timed context (e.g., see abstract).

Taylor (patent No. 5,497,498) disclosed a video processing module using a second programmable logic device which reconfigures a first programmable logic device for data transformation (e.g., see abstract).

Mirsky (patent No. 5,915,123) disclosed a system for controlling configuration memory contexts of processing elements in a network of multiple context processing elements (e.g., see abstract).

Fallside (patent No. 6,326,806) disclosed a FPGA-based communication access point and system for reconfiguration (e.g., see abstract).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Coleman whose telephone number is (571) 272-4163. The examiner can normally be reached on Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

EC

ERIC COLEMAN
PRIMARY EXAMINER